

# CloudSMS平台Web API接口文档\_v2.3\_中文版

## CloudSMS Web API 接口简介

### 概论

本文档用来说明 CloudSMS平台的Web API接口规范。CloudSMS平台是基于RESTful风格进行开发的一款短信服务平台，其采用通用的 UIP (Unified Interface Platform) 标准来规范各计算机系统之间的通信。

UIP 接口主要用在私有接口通信领域，比如业务的请求响应，账户的开通与关闭。

### 接口规范及通信原则

- 利用 JSON 规范封装 UIP 接口数据。有关 JSON 详情可以参考：<http://www.json.org/json-zh.html>
- 在 TCP 上传输，由通信双方协商确定通信端口。
- 是否需要长连接可根据具体业务来选择，UIP 定义专门的保持长连接的心跳机制。
- UIP 包为请求和应答一一对应的模式。
- 每一个 UIP 包中必须包含的如下四个方法字段：
  - METHOD, TYPE, SERIAL, TIME，详情请参考 [常用参数解释](#)。
- 通信双方需使用 HTTPS 进行加密，SSL 协商使用双向认证的方式。
- 所有请求均采用 HTTPS+POST 协议。
- HTTPS 消息体采用 JSON 格式。

### 名词解释及简写

- LMAX: 数据的最大字节长度；
- UINT: 无符号整型
- STR: 字符串
- Array: 数组

# 历史更新日志

版本号及发布时间	修改及内容说明
CloudSMS 2.0_2021_04_26	<ul style="list-style-type: none"><li>· 更新CloudSMS各环境访问地址信息</li><li>· 添加RESULT_CODE错误码说明及相应解决方法</li><li>· 添加LOCAL_REJECTED错误码说明</li><li>· 添加CloudSMS短信切片规则说明</li><li>· 新增Python请求脚本</li></ul>
CloudSMS 2.1_2021_04_27	<ul style="list-style-type: none"><li>· 新增群发能力（SMS_SEND_REQUEST）</li></ul>
CloudSMS 2.2_2021_06_15	<ul style="list-style-type: none"><li>· 修改DR回调报告格式</li></ul>
CloudSMS 2.3_2021_09_13	<ul style="list-style-type: none"><li>· 新增DCS3 8Bit的短信切片规则</li></ul>

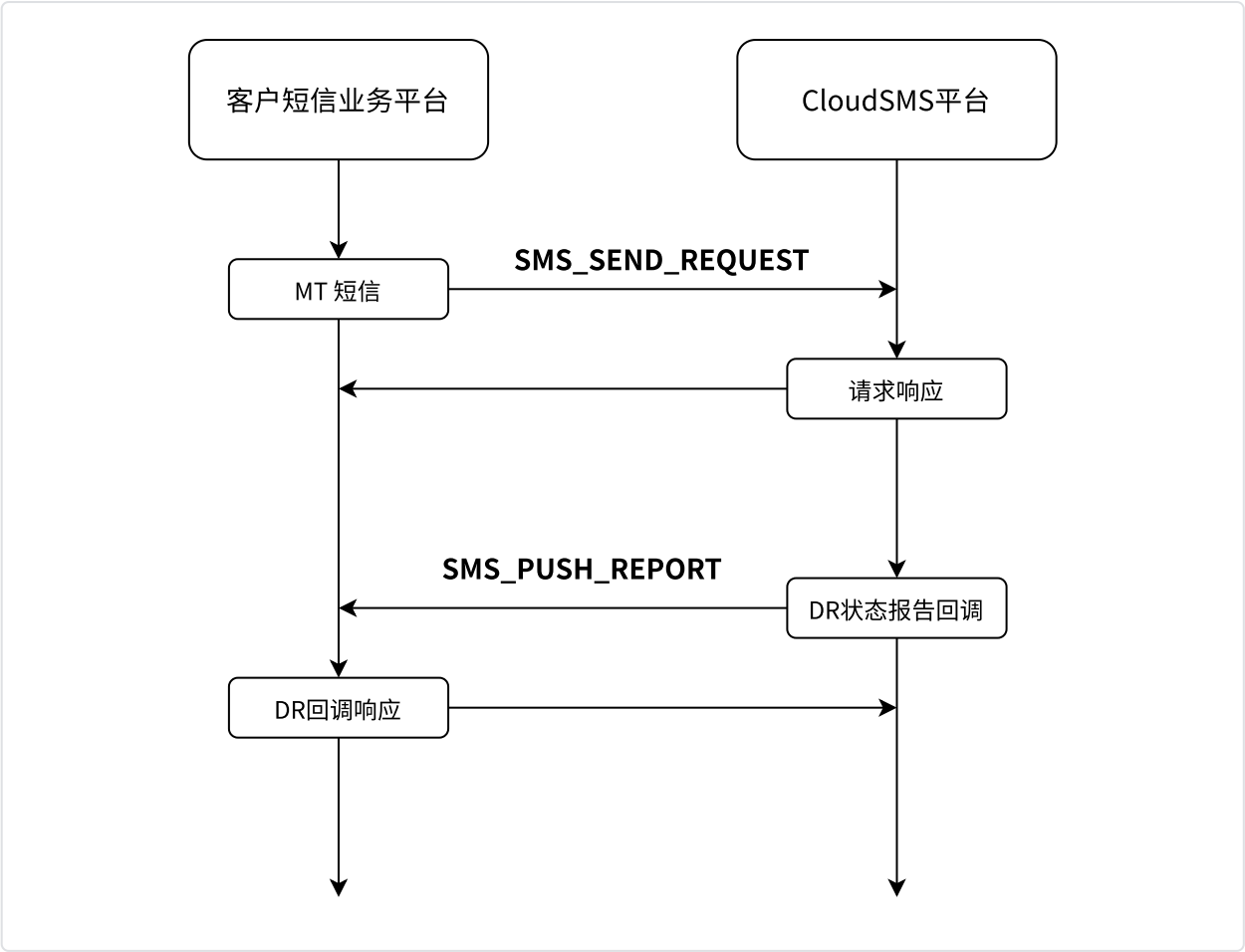
## CloudSMS平台服务器

为保障客户与CloudSMS平台之间数据通信的安全，CloudSMS平台将采用白名单限制的方式，藉此需收集客户的相关网络信息作为验证依据。

	项目	服务器链接	备注
短信发送请求	测试服务器	<a href="https://cloudsms-new.jegotrip.com.cn:61815/dave/io">https://cloudsms-new.jegotrip.com.cn:61815/dave/io</a>	需要客户提供源公网IP地址注册白名单
	生产服务器	<a href="https://cloudsms-new.jegotrip.com.cn:1815/dave/io">https://cloudsms-new.jegotrip.com.cn:1815/dave/io</a>	需要客户提供源公网IP地址注册白名单
短信状态回调	Cloud SM 回调发送服务器	223.119.32.232； 223.119.32.134 223.119.32.139； 223.119.32.250 223.119.32.160 (STG测试环境)	发送地址属于并列关系，将随机使用，客户需全部加入其服务器白名单。
	客户回调 对接服务器	请客户提交其对接服务器链接（仅支持一个回调地址，推荐使用HTTP/HTTPS链接）	用于记录客户接受短信回调信息链接。

# CloudSMS 业务流程

用户可使用其客户平台通过 Web API 接口（HTTPS）连接 CloudSMS平台。使用CloudSMS平台 提供的短信发送功能提交短信发送请求 UIP 数据包，CloudSMS平台 接收到请求并验证成功后向客户发送回应UIP 数据包。操作完成后向客户平台推送短信的状态信息，包含短消息发送状态、发送数量、费用信息等。



## 常用参数解释

每个 UIP 数据包中必须包含如下四个参数，参数详细解释如下表所示：

参数名称	数据类型	LMAX	说明	必须
METHOD	STR	-	描述该 UIP 所属业务。	是
TYPE	STR	8	表示该 UIP 的类型：请求或应答	是

SERIAL	UINT	4	用于记录 UIP 包的发送序号。	是
TIME	STR	32	记录 UIP 包发出的本地时间。	是

## METHOD 参数

用来描述该UIP所属的业务，提供两个值选择：

- SMS\_SEND\_REQUEST：表示请求短信发送操作；
- SMS\_PUSH\_REPORT：表示请求短信状态报告操作。

## TYPE 参数

表示该UIP的类型，提供两个值选择：

- REQUEST：表示为请求 UIP 消息的类型；
- ANSWER：表示为应答 UIP 消息的类型。

## SERIAL 参数

用于规范 UIP 数据包传输的顺序，由传输双方各自计数，从 1 开始到 128 进行循环计数。该字段的作用主要是用于容错，以防止存在有数据包没有被处理或系统异常的情况。（大于等于 1，小于等于 65535）

## TIME 参数

用于记录 UIP 数据包发出时的本地时间。

- 例如：北京时间（UTC+8），时间记录格式为：YYYY-MM-DD HH:MM:SS

## CloudSMS平台使用方法

CloudSMS平台 支持两种短信息请求功能，分别为短消息发送请求和短信状态报告推送。每一个请求 UIP 数据包都需要有一个相对应的应答包与之对应。

# 短信发送操作

## 操作方法介绍

当方法名称被设置为 **SMS\_SEND\_REQUEST** 时，表示通过该 UIP 包请求发送短消息。客户可在请求包中添加JSON格式的自定义参数。自定义参数将在回调DR中反馈。请求 UIP 包详细内容如下表所示。

## 短消息发送 UIP 的请求包格式

请求方式	POST
------	------

参数名称	数据类型	LMAX	说明	必须
METHOD	STR	-	SMS_SEND_REQUEST	是
TYPE	STR	8	REQUEST	是
SERIAL	UINT	4	取值 1~65535	是
TIME	STR	32	记录 UIP 包发出的本地时间。	是
AUTH_KEY	STR	128	固定长度为 128 字节的字符串。	是
MULTI_MSISDN_LIST	Array		用于携带电话号码数据。单次支持最多200个号码。	是
MULTI_MSISDN_LIST     数组内容开始				
COUNTRY_CODE	UINT	4	用于携带国家地区代码， 如：[中国大陆:86]	是
DEST_MSISDN	STR	32	用于携带目的电话的电话号码	是
MULTI_MSISDN_LIST     数组内容结束				
SMS_CONTENT	STR	1600	这个字段携带短信具体内容	是
ROUTE_ID	STR	255	表示CloudSMS平台的路由策略名称。 由相关运维人员配置具体策略生成。	是
ORIGINAL_ADDR	STR	32	用于携带短信发送号码的数字或字母	否
SIGNATURE	STR	64	携带短信的签名的字符串	否

<b>SIGNATURE_TYPE</b>	UINT	4	设置签名的格式与位置	否
<b>PRIORITY</b>	UINT	4	用于设置短消息的优先级。 <ul style="list-style-type: none"> <li>0:普通优先级(缺省方案)</li> <li>1:高优先级</li> <li>&gt;1:保留</li> </ul>	否
<b>VERSION</b>	STR	12	表示当前 CloudSMS 的版本 取值：2021-01-01	是
<b>CUSTOMER_BODY</b>	OBJ	255	储存客户自定义JSON结构数据	否

## 短消息发送 UIP 的应答包格式

每个 UIP 请求包都需要一个应答包与之一一对应，应答包如下表所示。

参数名称	数据类型	LMAX	说明	必须
<b>METHOD</b>	STR	-	SMS_SEND_REQUEST	是
<b>TYPE</b>	STR	8	ANSWER	是
<b>SERIAL</b>	UINT	4	取值 1~65535	是
<b>TIME</b>	STR	32	记录 UIP 包发出的本地时间。	是
<b>RESULT_CODE</b>	INT	4	应答结果码回复	是
<b>RESULT_DESC</b>	STR	127	返回结果码的描述信息ANSI 字符集中的可显示字符。	是
<b>DETAIL_LIST</b>	Array	--		是
<b>DETAIL_LIST 数组开始</b>				
<b>COUNTRY_CODE</b>	UINT	4	这个字段用于携带国家地区代码。 如：[中国大陆:86]	是
<b>DEST_MSISDN</b>	STR	32	短消息目的电话的电话号码。	是
<b>SMS_UID</b>	STR	32	32 位以内的数字字符串，作为短信的唯一标	是

			识	
RESULT_CODE	INT	4	应答结果码回复	是
RESULT_DESC	STR	127	返回结果码的描述信息	是
DETAIL_LIST 数组字段已结束				

备注：支持在 MULTI\_MSISDN\_LIST 字段中群发小批量短信，最大支持200条。每条短信都将获得一个 SMS\_UID。

## 短消息发送的参考示例

### 示例1 - 单条短信发送

#### 短信发送请求包（单条发送）

Python

```
1  {
2      "METHOD": "SMS_SEND_REQUEST",
3      "TYPE": "REQUEST",
4      "SERIAL": 1,
5      "TIME": "Please input your local time",
6      "AUTH_KEY": "Please input your auth_key here",
7      "ROUTE_ID": "Please input your Route_ID here",
8      "MULTI_MSISDN_LIST":
9      [
10         {
11             "DEST_MSISDN": "67xxxx11",
12             "COUNTRY_CODE": 852
13         }
14     ],
15     "SMS_CONTENT": "您的验证码是123345，请勿向他人透露。",
16     "SIGNATURE": "CloudSMS", # 短信签名无需填写【】，方框将通过SIGNATURE_TYPE添加
17     "SIGNATURE_TYPE": 1, # 如使用SIGNATURE参数，请配合选择SIGNATURE_TYPE字段
18     "ORIGINAL_ADDR": "", # 参数值若不填或为空，短信将使用默认SENDERID
19     "VERSION": "2021-01-01",
20     "REPEAT": 0,
21     "CUSTOMER_BODY": {}
22 }
```

#### 短信发送响应包（单条发送）

## Python

```
1  {
2      "METHOD": "SMS_SEND_REQUEST",
3      "TYPE": "ANSWER",
4      "SERIAL": 1,
5      "TIME": "响应时间", # 时间格式为: YYYY-MM-DD HH:MM:SS
6      "RESULT_CODE": 1, # RESULT_CODE详情请参考附录RESULT_CODE和RESULT_DESC参数
7      "RESULT_DESC": "'OK'",
8      "DETAIL_LIST":
9      [
10         {
11             "RESULT_CODE": 1,
12             "RESULT_DESC": "'OK'",
13             "DEST_MSISDN": "67xxxx11",
14             "COUNTRY_CODE": 852,
15             "SMS_UID": "5A3B394096E50X01D1FA163E81BFFF"
16         }
17     ]
18 }
```

### 示例2 - 小批量群发短信

#### 短信发送请求包（小批量群发）



## Python

```
1  {
2      "METHOD": "SMS_SEND_REQUEST",
3      "TYPE": "REQUEST",
4      "SERIAL": 1,
5      "TIME": "Please input your local time",
6      "AUTH_KEY": "Please input your auth_key here",
7      "ROUTE_ID": "Please input your Route_ID here",
8      "MULTI_MSISDN_LIST":
9      [
10         {
11             "DEST_MSISDN": "67xxxx11",
12             "COUNTRY_CODE": 852
13         },
14         {
15             "DEST_MSISDN": "67xxxx28",
16             "COUNTRY_CODE": 852
17         }
18     ],
19     "SMS_CONTENT": "您的验证码是123345，请勿向他人透露。",
20     "SIGNATURE": "CloudSMS", # 短信签名无需填写 【】，方框将通过SIGNATURE_TYPE添加
21     "SIGNATURE_TYPE": 1, # 如使用SIGNATURE参数，请配合选择SIGNATURE_TYPE字段
22     "ORIGINAL_ADDR": "", # 参数值若不填或为空，短信将使用默认SENDERID
23     "VERSION": "2021-01-01",
24     "REPEAT": 0,
25     "CUSTOMER_BODY": {}
26 }
```

短信发送响应包（小批量群发）

## Python

```
1  {
2      "METHOD": "SMS_SEND_REQUEST",
3      "TYPE": "ANSWER",
4      "SERIAL": 1,
5      "TIME": "响应时间", # 时间格式为: YYYY-MM-DD HH:MM:SS
6      "RESULT_CODE": 1, # RESULT_CODE详情请参考附录RESULT_CODE和RESULT_DESC参数
7      "RESULT_DESC": "'OK'",
8      "DETAIL_LIST":
9      [
10         {
11             "RESULT_CODE": 1,
12             "RESULT_DESC": "'OK'",
13             "DEST_MSISDN": "67xxxx11",
14             "COUNTRY_CODE": 852,
15             "SMS_UID": "5A3B394096E50X01D1FA163E81BFFF"
16         },
17         {
18             "RESULT_CODE": 1,
19             "RESULT_DESC": "'OK'",
20             "DEST_MSISDN": "67xxxx28",
21             "COUNTRY_CODE": 852,
22             "SMS_UID": "5A3B394096E50X02D1FA163E81BCCC"
23         }
24     ]
25 }
```

## 平台短信状态报告推送（回调）

CloudSMS支持向客户DR系统推送短信发送DR回执操作。客户需要建立一个HTTP/HTTPS服务器，用于接收来自CloudSMS平台推送的DR回执信息。

备注：DR回执最长返回时间为24小时。若CloudSMS在超过25小时后仍未收到落地运营商返回的DR报告，CloudSMS将认为该短信为超时状态，并回复 **EXPIRED** 状态的DR回执到客户DR系统。

## 操作方法介绍

当METHOD参数被设置为 **SMS\_PUSH\_REPORT** 时，表示CloudSMS平台将通过该POST请求包向客户短信DR服务器发送状态报告推送。UIP 包详细内容如下表所示。

## DR状态报告 UIP 的请求包格式

--	--

请求方式	POST
------	------

参数名称	数据类型	LMAX	说明	必须
METHOD	STR	--	SMS_PUSH_REPORT	是
AUTH_KEY	STR	128	固定长度为 128 字节的字符串。	是
TYPE	STR	8	REQUEST	是
SERIAL	UINT	4	取值 1~65535	是
TIME	STR	32	UTC+8时间 (香港时间)	是
SMS_UID	STR	32	短信的唯一标识	是
SOURCE_MSISDN	STR	32	携带短信源的电话号码	是
SMS_RESULT_LIST	Array	--	--	是
SMS_RESULT_LIST 数组内容				
COUNTRY_CODE	UINT	4	用于携带国家地区代码, 如: [中国大陆:86]	是
DEST_MSISDN	STR	32	携带短信目标的电话号码	是
SMS_STATE	UINT	4	短信的最终状态	是
RESULT_DESC	STR	32	对短信报告状态的解释	是
SMS_RESULT_LIST 数组内容结束				
REPORT_TIME	STR	32	提交报告的时间, UTC+8时间 (香港时间)	否
CUSTOMER_BODY	OBJ	255	携带客户自定义JSON结构数据	否

DR状态报告 UIP 的应答包格式

参数名称	数据类型	LMAX	说明	必须
METHOD	STR	-	SMS_PUSH_REPORT	是

TYPE	STR	8	ANSWER	是
SERIAL	UINT	4	取值 1~65535	是
TIME	STR	32	记录 UIP 包发出的本地时间	是
RESULT_CODE	INT	4	应答结果码回复	是
RESULT_DESC	STR	127	返回结果码描述信息	是

短消息状态报告推送的参考实例

DR回调请求包（由CloudSMS发起）

若使用小批量短信群发功能，DR回执仍会按照SMS\_UID单独推送至客户DR系统。

Python

```
1  {
2    "METHOD": "SMS_PUSH_REPORT",
3    "TYPE": "REQUEST",
4    "SERIAL": 2,
5    "TIME": "DR报告推送时间", # 时间格式为YYYY-MM-DD HH:MM:SS
6    "REPORT_TIME": "DR报告提交时间", # 时间格式为YYYY-MM-DD HH:MM:SS
7    "SOURCE_MSISDN": "CloudSMS",
8    "SMS_UID": "5A3B394096E50X01D1FA163E81BFFF",
9    "SMS_RESULT_LIST":
10   [
11     {
12       "COUNTRY_CODE": 852,
13       "DEST_MSISDN": "67xxxx11",
14       "SMS_STATE": 2,
15       "RESULT_DESC": "DELIVERED",
16     }
17   ],
18   "CUSTOMER_BODY": {}
19 }
```

DR回调响应包

## Python

```
1 {  
2     "METHOD": " SMS_PUSH_REPORT",  
3     "TYPE": "ANSWER",  
4     "SERIAL": 2,  
5     "TIME": "响应时间", # 时间格式为YYYY-MM-DD HH:MM:SS  
6     "RESULT_CODE": 1,  
7     "RESULT_DESC": "OK"  
8 }
```

## 参数详细解释

### AUTH\_KEY参数

该参数是一个固定长度为 128 字节的字符串。用于携带一个身份标识，使CloudSMS服务器可以通过该参数对用户进行识别。

例如： "AUTH\_KEY":

```
"c0UlfIimjcBgfx5cNai0Z7BLsonagmj t5+vncIGdK0sZXsjkuQjERKtSrm7iTvNKgcXvv  
LsK9+ptfYQHFgB0a=="
```

### SMS\_CONTENT参数

该参数用于携带短信内容，允许最大长度为1600字节，字节计算规则如下：

- 纯英文（标点符号全为半角）按一个字符一个字节。
- 其他皆为一个字符两个字节计算（中英混合，英文也为两个字节）。
- 根据SMPP协议，长短信会被CloudSMS平台切分计费。切片规则详情请参考 [附录 CloudSMS短信切片规则](#)

### ROUTE\_ID 参数

该参数表示CloudSMS平台的路由名称，由CloudSMS路由策略自动产生。

例如： "ROUTE\_ID": "VerifyCode\_516"

## SIGNATURE 参数

对于发往中国大陆或海外有监管要求的方向，需要预先在CloudSMS平台成功注册短信签名，例如[CloudSMS]。如需申请短信签名，请联系销售经理。

## SIGNATURE\_TYPE 参数

用于格式化签名的样式和在短信上的展示位置。详细参数如下表所示：

SIGNATURE_TYPE	类型	位置	说明
1	实心方头括号【】	短信最开头	【Signature】xxx...
2	实心方头括号【】	短信结尾	xxx... 【Signature】
3	方括号 []	短信最开头	[Signature] xxx...
4	方括号 []	短信结尾	xxx...[signature]
5	无	无	Signature 已经拼接完成，不设置签名位置

## CUSTOMER\_BODY参数

`CUSTOMER_BODY` 参数是用于携带客户自定义参数的字段。该参数会在DR报告中无修改的推送到客户DR系统。

例如：客户可分别为每条短信自定义分配识别代码，在DR回执中获取识别代码用于客户内部系统的短信识别。

## RESULT\_CODE 和 RESULT\_DESC 参数

结果码在回应包中使用，用于表示请求 UIP 包的接受结果。

RESULT_CODE	RESULT_DESC	说明	解决方法

0x01	OK	成功。	
0x02	Unknown_error	未知异常错误。	请向CloudSMS报障
0x03	Busy	平台忙碌中。	请稍后重试（10分钟）
0x04	Empty_data	空数据包。	请检查API请求数据包是否正确提交
0x05	Channel_not_exist	AUTH_KEY错误。	请检查AUTH_KEY是否填写正确
0x10000001	Invalid_parameter	参数错误。	请检查API请求包参数是否正确。 <i>例如：参数名称是否正确、数据类型是否正确</i>
0x10000003	Invalid_context	无效的短信内容，短信内容丢失或太长。	请检查API数据包是否为JSON格式； 检查API数据包必填参数是否完整。
0x10000004	Send_failed	短信发送失败。	请向CloudSMS报障
0x10000005	Queue_too_large	缓存区已满。	请稍后重试（10分钟）
0x10000006	Intercept	短信被拦截（可能为黑名单）。	请稍后重试（10分钟）
0x10000007	Lost_info	数据包不全（部分丢失）	检查API数据包必填参数是否完整。
0x1000000D	Empty_data	收到空数据包	请检查客户API数据包格式和必填参数是否完整。
0x1000000E	DB_not_find	数据存储错误	请稍后重试（10分钟）
0x1000000F	Unpriced_area	该地区不允许发送或未设置价格信息	请检查支持发送的短信方向。
0x10000010	Invalid_Version	无效的版本号	请更换为最新的Web API 版本号。

以下情形将会出现 `Invalid_parameter (0x10000001)` 的错误码，请参考：

- 必填参数缺失或未填写
- SOURCE\_MSISDN 参数的字符值不在ASSIC码0x20 – 0x80的范围内。
- 无效的国家码
- DEST\_MSISDN参数的值为非数字或长度超过32字节

- 发送至中国大陆方向的号码长度不是11位

## SMS\_STATE 和 RESULT\_DESC 参数

状态码在平台短信状态推送操作中用于指示短消息的传输状态。

SMS_STATE	RESULT_DESC	说明
1	ENROUTE	消息正在传输
2	DELIVERED	消息已送达
3	EXPIRED	消息超时
4	DELETED	消息被删除
5	UNDELIVERABLE	消息不可送达
6	ACCEPTED	消息已接收（例如：已经被客服代表用户手工读取）
7	UNKNOWN	消息状态无效
8	REJECTED	消息被拒绝
9	IDLE	消息正在处理中
60	LOCAL_REJECTED	短信被CloudSMS平台本地拒绝

以下情况将会产生错误代码LOCAL\_REJECTED（60）：

- 无效的新加坡号码判断规则：
  - 新加坡电话号码开头不是8或9
  - 新加坡电话号码不是八位（不包括国家码）
- 未配置的发送方向（不包括在产品明细中的发送方向）
- SMS\_SEND\_REQUEST中使用了错误的ROUTE\_ID参数值
- 账户余额不足

## 附录



# CloudSMS短信切片规则

Standard English Characters (GSM DCS0 7bit encoding)	Segment Quantity(s)
1 – 160 characters	1
161 – 306 characters	2
307 – 459 characters	3
460 – 612 characters	4
613 – 765 characters	5
766 – 918 characters	6
919 – 1071 characters	7
1072 – 1224 characters	8
1225 – 1377 characters	9
1378 – 1530 characters	10

Latin 1 (ISO-8859-1) (DSC3 8bit encoding)	Segment Quantity(s)
1 – 140 characters	1
141 - 268 characters	2
269 - 402 characters	3
403 - 536 characters	4
537 - 670 characters	5
671 - 804 characters	6
805 - 938 characters	7

938 - 1072 characters	8
1073 - 1206 characters	9
1207 - 1340 characters	10

Non-GSM (Unicode) characters (DCS8 16bit encoding)	Segment Quantity(s)
1 – 70 characters	1
71 – 134 characters	2
135 – 201 characters	3
202 – 268 characters	4
269 – 335 characters	5
336 – 402 characters	6
403 – 469 characters	7
470 – 536 characters	8
537 – 603 characters	9
604 – 670 characters	10

## 验证码常见格式

【CloudSMS】您的短信验证码是：123456。请勿告诉任何人，任何索取行为均为诈骗！不要在任何短信或邮件链接页面输入验证码。
[CloudSMS] Your Verification Code: 123456. Please do not disclose this code to anyone. Anyone request for this code is recognized as a fraud. Please do not include this code in any short message or email.

## 支持结算币种

--	--

货币名称	货币简写
人民币	CNY
港币	HKD
台币	TWD
欧元	EUR
美元	USD
英镑	GBP
澳元	AUD
韩元	KRW
日元	JPY

测试脚本代码参考案例

Python

Python

```
1  import tornado.web
2  import tornado.ioloop
3  import requests
4  import time
5  import json
6
7  # 获取当前时间
8  local_time = time.strftime("%Y-%m-%d %H:%M:%S", time.localtime())
9
10 # 请求地址, 请从接口文档获取真实链接地址
11 url = ""
12
13 data = {
14     "METHOD": "SMS_SEND_REQUEST",
15     "TYPE": "REQUEST",
16     "SERIAL": 1,
17     "TIME": local_time,
```

```

18     "AUTH_KEY": "填写CloudSMS提供的AUTH_KEY参数",
19     "MULTI_MSISDN_LIST":
20     [
21         {
22             "DEST_MSISDN": "67xxxx11", # 接收的电话号码
23             "COUNTRY_CODE": 852, # 接受电话号码的国家码
24         }
25     ],
26     "SMS_CONTENT": "请填写您的短信内容",
27     "ROUTE_ID": "请填写CloudSMS提供的ROUTE_ID参数",
28     "PRIORITY": 0,
29     "SIGNATURE": "CloudSMS", # 如发往中国内地或海外存在监管需求的方向, 需提前向CloudS
    MS申请
30     "SIGNATURE_TYPE": 3,
31     "ORIGINAL_ADDR": "CloudSMS", # 非必填项。若未填写或值为空, 则会使用默认值CloudSMS
32     "VERSION": "2021-01-01", # 请使用当前最新版本
33 }
34
35 data = json.dumps(data)
36 print("Json: ", data)
37
38 class IndexHandler(tornado.web.RequestHandler):
39     def post(self):
40         jsonbyte = self.request.body
41         jsonstr = jsonbyte.decode('utf-8')
42         print(jsonstr)
43
44     rsp = requests.post(url, data, verify=False).content.decode('utf-8')
45     print("rsp: ", rsp)
46
47
48 if __name__ == "__main__":
49     app = tornado.web.Application(
50         [(r'/', IndexHandler)], static_path='static', debug=True, autoreload=False)
51     app.listen(59999)
52     tornado.ioloop.IOLoop.current().start()

```

## Ruby

## Ruby

```
1 require 'net/http'
2 require 'json'
3
4 def send_post(url,toSend)
5   uri = URI(url)
6   http = Net::HTTP.new(uri.host, uri.port)
7   if uri.scheme == "https"
8     http.use_ssl = true
9     http.verify_mode = OpenSSL::SSL::VERIFY_NONE
10  end
11
12  req = Net::HTTP::Post.new(uri.path, initheader = {'Content-Type' =>'application/json'})
13  req.body = toSend
14  res = http.request(req)
15  return JSON.parse res.body
16 end
17
18 data = {
19   "METHOD": "SMS_SEND_REQUEST",
20   "TYPE": "REQUEST",
21   "SERIAL": 1,
22   "TIME": "请填写发送时间", # 格式: YYYY-MM-DD HH:MM:SS
23   "AUTH_KEY": "填写CloudSMS提供的AUTH_KEY参数",
24   "MULTI_MSISDN_LIST":
25   [
26     {
27       "DEST_MSISDN": "67xxxx11",
28       "COUNTRY_CODE": 852,
29     }
30   ],
31   "SMS_CONTENT": "请填写您的短信内容",
32   "ROUTE_ID": "请填写CloudSMS提供的ROUTE_ID参数",
33   "PRIORITY": 0,
34   "SIGNATURE": "CloudSMS",
35   "SIGNATURE_TYPE": 3,
36   "VERSION": "2021-01-01",
37 }.to_json
38
39 tester = send_post("CloudSMS Server", data)
40 puts tester
```

# Node.js

## JavaScript

```
1  const https = require("https");
2
3  const data = JSON.stringify({
4    METHOD: "SMS_SEND_REQUEST",
5    TYPE: "REQUEST",
6    SERIAL: 1,
7    TIME: "请填写发送时间", # 格式: YYYY-MM-DD HH:MM:SS
8    AUTH_KEY: "填写CloudSMS提供的AUTH_KEY参数",
9    MULTI_MSISDN_LIST: [
10     {
11       DEST_MSISDN: "67xxxx11",
12       COUNTRY_CODE: 852,
13     },
14   ],
15   SMS_CONTENT: "请填写您的短信内容",
16   ROUTE_ID: "填写CloudSMS提供的AUTH_KEY参数",
17   PRIORITY: 0,
18   SIGNATURE: "CloudSMS",
19   SIGNATURE_TYPE: 3,
20   VERSION: "2021-01-01",
21 });
22
23 # 请求地址, 请从接口文档获取真实链接地址
24 const options = {
25   hostname: "",
26   port: 1815,
27   path: "/dave/io",
28   method: "POST",
29   headers: {
30     "Content-Type": "application/json",
31     "Content-Length": data.length,
32   },
33 };
34
35 const req = https.request(options, (res) => {
36   console.log(`statusCode: ${res.statusCode}`);
37
38   res.on("data", (d) => {
39     process.stdout.write(d);
```

```

39     process.stdout.write(a);
40   });
41 });
42
43 req.on("error", (error) => {
44   console.error(error);
45 });
46
47 req.write(data);
48 req.end();

```

## Java – RestTemplate

Python

```

1  package com.example.demo;
2
3  import org.json.JSONObject;
4  import org.springframework.boot.SpringApplication;
5  import org.springframework.boot.autoconfigure.SpringBootApplication;
6  import org.springframework.http.HttpEntity;
7  import org.springframework.http.HttpHeaders;
8  import org.springframework.http.MediaType;
9  import org.springframework.http.ResponseEntity;
10 import org.springframework.web.client.RestTemplate;
11
12 @SpringBootApplication
13 public class DemoApplication {
14
15     public static void cloudSmsPost() {
16
17         RestTemplate template = new RestTemplate();
18         // Request URL
19         String url = ""; # 请求地址, 请从接口文档获取真实链接地址
20
21         // JSON Data
22         JSONObject msisdn_list = new JSONObject();
23         msisdn_list.put("DEST_MSISDN", "67xxxx11");
24         msisdn_list.put("COUNTRY_CODE", 852);
25
26         JSONObject param = new JSONObject();
27         param.put("METHOD", "SMS_SEND_REQUEST");
28         param.put("TYPE", "REQUEST");

```

```

29     param.put("SERIAL", 1);
30     param.put("TIME", "请填写发送时间");
31     param.put("AUTH_KEY", "填写CloudSMS提供的AUTH_KEY参数");
32     param.put("SMS_CONTENT", "请填写您的短信内容");
33     param.put("ROUTE_ID", "填写CloudSMS提供的AUTH_KEY参数");
34     param.put("ORIGINAL_ADDR", "CloudSMS");
35     param.put("PRIORITY", 0);
36     param.put("SIGNATURE", "CloudSMS");
37     param.put("SIGNATURE_TYPE", 3);
38     param.put("VERSION", "2021-01-01");
39     param.put("MULTI_MSISDN_LIST", msisdn_list);
40     // POST Headers
41     HttpHeaders headers = new HttpHeaders();
42     headers.setContentType(MediaType.APPLICATION_JSON);
43     HttpEntity<String> entity = new HttpEntity<String>(headers);
44     ResponseEntity<String> result = template.postForEntity(url, entity, String.class);
45
46     // POST
47     String body = entity.getBody(); // get respond body
48     System.out.println("Param: " + param.toString());
49     System.out.println("Respond_body: " + body);
50     System.out.println("Respond_result: " + result);
51 }
52
53 public static void main(String[] args) {
54     SpringApplication.run(DemoApplication.class, args);
55     DemoApplication.cloudSmsPost ();
56 }
57
58 }

```